

REMARKS/ARGUMENTS

This Amendment is in response to the Final Office Action dated December 3, 2003.

Claims 1-59 remain pending in the present application. Claims 1, 18, 30-34, 38, 44-47, 51, and 57-59 are rejected. Claims 30 and 44 have been amended to include minor corrections for clarity and to present the claims in better form for appeal, if necessary, and these amendments introduce no new issues for consideration.

The 102 Rejections

The Examiner rejected claims 30, 31, 44, 45, 57, and 58 under 35 U.S.C. 102(b) as being anticipated by Beier et al. '820 ("Beier"). Claim 30 recites a method for reorganizing a database table online, allowing the database table to be scanned, accessed, and updated during reorganization, and including a vacate move step to move data records from move pages in the table to available space in the database table, and a fill move step to move data records into move pages in the table, where the database table can be accessed, scanned, and updated during the vacate and fill move steps of the reorganization.

The Examiner states that Beier discloses these features at locations particularly cited by the Examiner. Applicant respectfully traverses, as discussed below.

The Examiner states that Beier teaches at col. 3, lines 37-60 a multi-step reorganization process, where the steps include running a pre-reorganization utility that examines structures, running a scan utility for data that is not being reorganized but is related to data that is being reorganized, including a reload operation and also running a prefix resolution utility and a prefix update utility. The Examiner states that Beier teaches scan utilities that are run on one or more related databases which are being pointed into or from the database being reorganized.

The Examiner apparently believes that these described “scan utilities” of Beier read on the scanning of the database table during reorganization as recited in claim 30. However, these scan utilities of Beier are not the same as the scanning recited in claim 30. Beier specifically states that the scan utility is run for data that is not being reorganized. These scan utilities are used to determine which other data elements are referenced by the data being reorganized. This is in direct contrast to claim 30, which recites that a database table is being reorganized, and that the database table itself, i.e. data that is being reorganized, can be scanned during the move steps of the reorganization. The cited lines of col. 3 in Beier do not disclose or suggest such scanning of database data that is being reorganized.

Furthermore, the Examiner states that Beier teaches at col. 5, lines 2-9 that when a database is being organized that has alternate, i.e., secondary, indexes associated with a data element being moved, at the time the data element is being moved from the old location to the new location, there is an ability to update all of the indexes.

The Examiner apparently believes that this described “updating” of indexes in Beier reads on the “updating” of the database table during reorganization as recited in claim 30. However, this updating of indexes by Beier is not the same as the updating recited in claim 30. Beier specifically states in the cited lines that secondary indexes, associated with data elements being moved, are being updated. Such indexes are pointers to the data elements being reorganized, and the update is the changing of the pointers to reflect the new location of the data elements. Updating these indexes are not the same as updating a database table—a database table is the actual data records in the database, i.e. the data elements. Applicant’s claim 30 thus recites that the data records themselves can be updated during reorganization, e.g., adding data records to the database table, deleting data records from the table, or modifying data records, by users or applications, in an online fashion, as indicated in Applicant’s specification throughout (e.g., page 1, lines 13-15, 19-22; page 12, lines 15-

19; page 13, lines 14-16; page 20, lines 5-7; page 26, lines 5-9). This is not the same as updating the indexes described by Beier. There is no teaching in Beier for updating data records of a database table during move steps of the reorganization as recited by Applicant. Similarly, the "prefix update utility" mentioned by Beier at col. 3, line 46 refers to a prefix, which is a group of pointers, not data records of a database table.

The Examiner states that a prior art disclosure need not be express in order to anticipate, and that even if a prior art inventor does not recognize a function of his or her process, the process can anticipate if that function was inherent or necessarily present as made clear by extrinsic evidence. However, there is no indication that Beier's invention would inherently allow features of Applicant's claim, such as, for example, updates made to a database table while that table is being reorganized, as recited by Applicant. Such a feature is not inherent to Beier, since it is described and suggested nowhere in Beier as discussed above, and further, Beier states that during reorganization, data elements of a data set are copied or "recreated" in a different portion or new data set (col. 15, lines 15-20, 24-27), which indicates that the data is being reorganized into a new database structure and storage. This type of reorganization does not allow Applicant's online scans or updates of data in a database table during reorganization, as further indicated in Beier at col. 11, lines 42-45, which states that the rest of the database "is available for use," while the reorganized portion is not available for use; or col. 4, lines 3-7. This is because an update of Beier's database table during reorganization would not be assured of correctly finding/updating recently moved or reorganized data elements. There is thus no indication of the features of Applicant's claims being inherent to Beier. Furthermore, the Examiner has cited no extrinsic evidence that shows the above-described features of claim 30 to be inherent to Beier's disclosure.

Applicant recites "online" reorganization in claim 30, i.e., the data records of the database table can be scanned, accessed, and updated during the reorganization, i.e., while move steps of the

reorganization are occurring. As discussed above, Beier does not disclose or suggest providing such online reorganization of the database table. Beier does not address, for example, scanning operations and updates during reorganization of the partition being reorganized because the Beier method either makes the data set being reorganized unavailable to scanners, as indicated in col. 7, lines 26-30 and col. 11, lines 42-45; or Beier must synchronize the incrementing of a "reorganization number" with the absence of scanners. Nowhere in Beier is a scanning or update of records suggested prior to, after, and across move steps of a reorganization process.

The Examiner also stated that Beier teaches a vacate move step to move data records from move pages in the table to available space in the database table at col. 5, lines 1-6. These cited lines mention only that a database element is being moved from an old location to a new location. As described above, Beier describes his process in which, during a reorganization, data elements of a data set are copied or "recreated" in a different partition or new data set (col. 15, lines 15-20, 25-27).

Beier's method of copying data elements from an old set to a new, reorganized data set is not the same as moving elements from and to the same stored database table, as recited by Applicant's vacate move step of claim 30. Similarly, Applicant's fill move step moves data records into the move pages in the same stored table; Beier's method, however, moves data elements to a new data set, not into the same data set from which data records were originally stored. Furthermore, Beier nowhere mentions the combination of vacate step and a fill step as in claim 30, where data is moved out of locations in the database table in the vacate step, and other data from the database table is then moved into the vacated locations in the fill step. Beier mentions only the generic moving of records from an old location to a new location at the cited lines.

For all the above reasons, claim 30 is therefore believed patentable over Beier. Claim 31 is dependent on claim 30 and is therefore patentable for at least the same reasons as claim 30, and for additional reasons.

As to claim 44, Beier does not disclose or suggest an online database table that can be scanned, accessed and updated during move steps, nor a vacate move step to move data records from move pages in the table to available space in other pages in the same stored table, nor a fill step to move data records into the move pages. Claim 44 is therefore patentable over Beier for reasons similar to claim 30. Claim 45 is dependent on claim 44 and is patentable over Beier for at least the same reasons as claim 44, and for additional reasons.

As to claim 57, Beier does not disclose or suggest online scanning, accessing and updating during reorganization, nor a vacate move step to move data records from move pages in the table to available space in other pages in the same stored table, nor a fill step to move data record into the move pages. Claim 57 is therefore patentable over Beier for reasons similar to claims 30 and 44. Claim 58 is dependent on claim 57 and is patentable over Beier for at least the same reasons as claim 57, and for additional reasons.

In view of the foregoing, Applicant believes that Beier does not teach or suggest the recited features of claims 30, 31, 44, 45, 57, and 58, and respectfully requests that the rejection under 35 U.S.C. 102 be withdrawn.

The 103 Rejections

The Examiner rejected claims 1, 18, 32-34, 38, 46, 47, 51, and 59 under 35 U.S.C. 103(a) as being unpatentable over Beier. Applicant respectfully traverses. Claim 1 recites a method for reorganizing a database table online, allowing the database table to be scanned, accessed, and updated during reorganization, and including moving a subset of records within the database table, flagging each moved record as a reorganization record, creating a reorganization pointer record at the initial location of the moved record, and establishing scanner process constraints, where the scanner process can correctly retrieve records from the database table during the

reorganization of the database table, including before and after the reorganizational movement of records.

The Examiner stated that Beier teaches the moving of a subset of records within the database table at col. 5, lines 1-6. However, as explained above, Beier only teaches moving data elements from one location to a new location at the cited lines, and does not teach or suggest moving data elements within a stored database table; rather, Beier's invention describes moving data elements from one data set to a completely new data set, as described above. Beier therefore does not disclose or suggest the recited element.

The Examiner stated that Beier teaches flagging each moved record as a reorganization record at col. 6, lines 55-60. These cited lines of Beier describe a "given segment" pointing to a "target segment," where the given segment has a context and reorganization number that indicates the reorganization level of the target at the time the direct pointer is assigned. Beier compares a pointer's reorganization number with a reorganization number assigned to the entire data partition to check whether the pointer is valid or out-of-date (col. 6, line 63 to col. 7, line 8). Beier's reorganization number thus is not a flag for a record moved during reorganization, but is a type of timestamp that indicates in which previous reorganization occurrence that the pointer to the target segment was updated. In contrast, claim 1 recites that a record that is moved during the current reorganization is flagged as a reorganization record. Beier's reorganization number is not a flag indicating whether a record is moved during a current reorganization, but rather indicates when a direct pointer to a segment was last updated in the past, and thus is not similar to Applicant's flagging step.

The Examiner stated that Beier teaches in the abstract "creating" a reorganization pointer record for each moved record at the initial location of the moved record, the reorganization pointer record pointing to the new location of the moved record. However, the abstract describes

Beier's process in which, for given data elements that point to a target data element, a direct pointer to the target data element is assigned at the time a given data element is created (abstract, lines 13-19). That direct pointer is used to reference the target element unless the pointer is found to be out of date, at which point an indirect pointer is used and the direct pointer is updated. In contrast, Applicant's claim 1 recites that a reorganization pointer record is created for each record moved during the reorganization and is stored at the initial location of the moved record. Applicant creates a pointer record at the time of reorganization for a record moved during reorganization, not at data element creation as in Beier. Applicant's pointer record allows old table scanners scanning during a reorganization to scan the database table, find the reorganization pointer record in the table where the moved record used to be, and follow the pointer and find the moved record accurately, as explained in Applicant's specification on page 20, lines 14-23, for example. In contrast, Beier's direct pointer is a pointer between existing data elements that is always present with the data elements, is not created during a reorganization, and is updated if it is found to be out of date. Beier's described steps are used to save reorganization steps so that a separate step of updating all the direct pointers is not needed (see abstract, lines 5-11), and do not themselves allow accurate scans of a database table during a reorganization as does Applicant's reorganization pointer record.

Furthermore, claim 1 recites that a reorganization pointer record is created for each moved record at the initial location of the moved record. In contrast, Beier installs a pointer, not a pointer record, to the new record location in a locator file ("Indirect List Entry (ILE)") (col. 5, line 65 to col. 6, line 4) and uses a reorganization number to select between the old and new pointers in the locator file (col. 14, lines 17-38). There is no disclosure or suggestion in Beier's abstract or elsewhere in Beier to create a reorganization pointer record in the database table at the initial location of the moved record, as stated by the Examiner.

The Examiner stated that it would have been obvious to a person of ordinary skill in the art to incorporate a scanner process to Beier teaching in order to improve data availability, citing col. 3 lines 37-46 and col. 15, lines 37-50. However, as discussed above for claim 30, the "scan utilities" mentioned by Beier at col. 3 are not used for scans of the database table during reorganization, but are utilities used on data that is not being reorganized. Claim 1 in contrast recites that the scanner process can retrieve records from the database table during the reorganization of that database table, i.e. data that is being reorganized. The lines at col. 15 of Beier describe the comparing of reorganization numbers, as discussed above, to determine if direct pointers are out of date based on the last update of the pointers during a previous reorganization, and mention nothing about scanner process constraints based on when a scan commences relative to the moving of data during a current reorganization process.

In addition, the multi-step reorganization process that Beier mentions at col. 3, lines 37-40 and col. 4, lines 1-7 is described as keeping the database unavailable for a long period of time, thus indicating that scanning processes of the database partition are not online and not allowed during the reorganization process of that partition. Beier nowhere mentions how such online scanning would continue from the old data set into Beier's reorganized, new data set, as discussed above for claim 30. Claim 1, in contrast, recites that a scanner process can correctly retrieve records from the database table during reorganization of that table.

Claims 34 and 47 recite features similar to those of claim 1, and are therefore patentable over Beier for at least similar reasons.

Beier therefore does not disclose or suggest the elements of claims 1, 34, and 47, and these claims are not obvious in view of Beier's disclosure.

Claim 18 recites a method for reorganizing a database table online, allowing the database table to be scanned, accessed and updated during the reorganization, including a vacate move

step, a vacate clean up step, a fill move step, and a fill clean up step. With respect to claims 18, 38, and 51, the Examiner stated that Beier discloses a vacate move step and fill move step at col. 5, lines 1-6; but these lines do not describe the cited move steps as similarly discussed above with reference to claim 1. The Examiner stated that Beier discloses a vacate clean up step at col. 1, lines 36-39. However, these cited lines merely describe a general technique of clustering database data to maximize performance, and the loss of clustering due to the deletion and adding of data elements over time during normal database operation (not reorganization). There is no mention of a specific vacate clean up step that removes temporary pointers provided in a vacate move step of a reorganization process as recited in claim 18.

The Examiner also stated that it would have been obvious to a person of ordinary skill to incorporate a scanner process to Beier teaching in order to improve data availability, citing col. 3, lines 37-46 and col. 15, lines 37-50, and that the fill clean up step of claim 18 is obvious. However, as discussed above for claim 30, the "scan utilities" mentioned by Beier at col. 3 are not used for scans of the database table during reorganization, but are utilities used on data that is not being reorganized. Claim 18 in contrast recites clean up and move steps synchronized to commence based on queries launching scanner processes of the database table, i.e., the scanner processes can occur during the reorganization of the database table. Furthermore, col. 3, lines 37-40 of Beier indicate that the database is unavailable during the multi-step reorganization process (database is only ready to be used after reorganization), such that no queries launching scanner processes are described during reorganization as recited in claim 18. It would not be obvious to include a scanner process during reorganization in Beier, as explained above, e.g., because Beier gives no suggestion how to continue a scanning process from the old data set into his reorganized, new data set.

The lines at col. 15 of Beier describe the comparing of reorganization numbers, as

discussed above, to determine if direct pointers are out of date based on the last update of the pointers during a previous reorganization, and mention nothing about the commencement of queries launching scanner processes, or of move steps, relative to the moving, clean-up, and queries of data during a current reorganization process as recited in claim 18.

Claims 38 and 51 recite features similar to those of claim 18 and are believed patentable for at least similar reasons to claim 18.

Claims 32, 33, 46, and 59 are patentable for at least the same reasons as their parent claims, which are discussed above, and for additional reasons. Claim 33, for example, recites overflow pointer records and the original position of a moved record in the database table, which are not described in the abstract as the Examiner stated; for example, the abstract describes direct pointers which point to a record directly, not to an original position of a moved record.

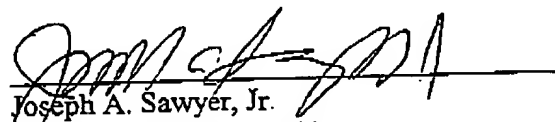
The cited references to Beier therefore do not disclose or suggest the features of Applicant's claims 1, 18, 32-34, 38, 46, 47, 51, and 59. Applicant therefore respectfully requests that the rejection of these claims under 103(a) be withdrawn.

In view of the foregoing, Applicant submits that claims 1-59 are patentable, and respectfully requests reconsideration and allowance of the claims as now presented.

Applicants' attorney believes this application in condition for allowance. Should any unresolved issues remain, the Examiner is invited to call Applicants' attorney at the telephone number indicated below.

February 3, 2004
Date

Respectfully submitted,
SAWYER LAW GROUP LLP


Joseph A. Sawyer, Jr.
Attorney for Applicant(s)
Reg. No. 30,801
(650) 493-4540